

Please type a plus sign (+) inside this box → ☐

02-25-00

PTO/SB/05 (12/97)

Approved for use through 9/30/00. OMB 0651-0032

Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

# UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No. ST9-99-153

Total Pages 48

First Named Inventor or Application Identifier

Yudong Sun

Express Mail Label No. EL487542388US

## APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

1. ☒ Fee Transmittal Form  
(Submit an original, and a duplicate for fee processing)
2. ☒ Specification [Total Pages 31]  
(preferred arrangement set forth below)
  - Descriptive title of the Invention
  - Cross References to Related Application
  - Statement Regarding Fed sponsored R & D
  - Reference to Microfiche Appendix
  - Background of the Invention
  - Brief Summary of the Invention
  - Brief Description of the Drawings (if filed)
  - Detailed Description
  - Claim(s)
  - Abstract of the Disclosure
3. ☒ Drawing(s) (35 USC 113) [Total Sheets 6]
4. Oath or Declaration [Total Pages 2]
  - a. ☒ Newly executed (original or copy)
  - b. ☐ Copy from a prior application (37 CFR 1.63(d))  
(for continuation /divisional with Box 17 completed)  
[Note Box 5 below]
    - i. ☐ DELETION OF INVENTOR(S)  
Signed statement attached deleting inventor(s)  
named in prior application, see 37 CFR  
1.63(d)(2) and 1.33(b).
5. ☐ Incorporation by Reference (useable if Box 4b is checked)  
The entire disclosure of the prior application, from which  
a copy of the oath or declaration is supplied under Box  
4b is considered as being part of the disclosure of the  
accompanying application and is hereby incorporated by  
reference therein.

6. ☐ Microfiche Computer Program (Appendix)
7. Nucleotide and/or Amino Acid Sequence Submission (if  
applicable, all necessary)
  - a. ☐ Computer Readable Copy
  - b. ☐ Paper Copy (identical to computer copy)
  - c. ☐ Statement verifying identify of above copies

## ACCOMPANYING APPLICATION PARTS

8. ☒ Assignment Papers (cover sheet & document(s))
9. ☐ 37 CFR 3.73(b) Statement ☐ Power of Attorney  
(when there is an assignee)
10. ☐ English Translation Document (if applicable)
11. ☒ Information Disclosure ☒ Copies of IDS  
Statement (IDS)/PTO-1449 Citations
12. ☐ Preliminary Amendment
13. ☒ Return Receipt Postcard (MPEP 503)  
(Should be specifically itemized)
14. ☐ Small Entity ☐ Statement filed in prior application,  
Statement(s) Status still proper and desired
15. ☐ Certified Copy of Priority Document(s)  
if foreign priority is claimed
16. ☒ Other: Express Mail Certificate

17. If a CONTINUING APPLICATION, check appropriate box and supply the requisite information:

☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No.:

## 18. CORRESPONDENCE ADDRESS

☒ Customer Number or Bar Code Label

21552

or ☐ Correspondence address below

NAME

Kory D. Christensen

ADDRESS

CITY

STATE

ZIP

COUNTRY

TELEPHONE

FAX

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, D.C. 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, D.C. 20231

PATENT APPLICATION

Docket No.: ST9-99-153



CERTIFICATE OF MAILING BY "EXPRESS MAIL"

"Express Mail" Mailing Label No.: EL487542388US

Date of Deposit: February 24, 2000

I hereby certify that this patent application in the name of Yudong Sun for SERVER-SIDE HTML CUSTOMIZATION BASED ON STYLE SHEETS AND A TARGET DEVICE, together with the drawings, a Utility Application Cover Sheet, a single signature Declaration, Power of Attorney, and Petition, an Assignment and Recordation Form Cover Sheet, Information Disclosure Statement, PTO Form 1449, Cited References, Fee Transmittal, and Check No. 13096 for \$950.00 are being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 C.F.R. § 1.10 on the date indicated above in an envelope addressed to Box Patent Application, Assistant Commissioner for Patents, Washington, D.C. 20231.

Respectfully submitted,

Kory D. Christensen

Reg. No.

Attorney for Applicant

Date: February 24, 2000

MADSON & METCALF

Gateway Tower West

15 West South Temple, Suite 900

Salt Lake City, Utah 84101

Telephone: 801/537-1700

c:\corr\000224 04

Express Mailing Label No. EL487542388US

PATENT APPLICATION

Docket No. 3000.2.28

IBM No. ST9-99-153

UNITED STATES PATENT APPLICATION

of

YUDONG SUN

for

SERVER-SIDE HTML CUSTOMIZATION BASED ON STYLE SHEETS AND A  
TARGET DEVICE

MADSON & METCALF, P.C.  
ATTORNEYS AT LAW  
900 GATEWAY TOWER WEST  
15 WEST SOUTH TEMPLE  
SALT LAKE CITY, UTAH 84101

# SERVER-SIDE HTML CUSTOMIZATION BASED ON STYLE SHEETS AND A TARGET DEVICE

## BACKGROUND OF THE INVENTION

### Field of the Invention

The present invention relates generally to techniques for processing a hypertext markup language (HTML) document. More particularly, the present invention relates to a system and method for server-side HTML customization based on style sheets and a target device.

### Identification of Copyright

A portion of the disclosure of this patent document contains material subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever.

### Relevant Technology

The World Wide Web (hereinafter "the Web") is a collection of Internet-accessible servers from which specially formatted documents may be retrieved and displayed by Web browsers, such as Netscape Navigator™ and Microsoft Internet Explorer™. Currently, the hypertext markup language ("HTML") is the most

1 common authoring language for creating Web documents, also known as Web  
2 pages. A Web page is identified by a uniform resource locator ("URL"), which is  
3 used by a Web browser to locate and display a particular Web page.

4 Web browsers are now found in a variety of target devices, some of which  
5 are not capable of displaying every possible Web page. For example, a personal  
6 data assistant (PDA) is a handheld device that often includes a Web browser.  
7 However, a PDA is typically limited to displaying a few lines of text, and may not  
8 be able to display images or other graphical objects. As such, specially modified  
9 Web pages are typically required for PDAs.

10 In addition, some target devices have bandwidth limits for accessing the  
11 Internet. Wireless devices, for instance, such as Web-enabled cellular phones, are  
12 not capable of rapidly processing large Web pages. Accordingly, specially  
13 modified versions of Web pages are also desirable in the context of limited-  
14 bandwidth target devices.

15 Unfortunately, providing target device-specific versions of Web pages  
16 usually means providing separate Web pages identified by different URLs, which  
17 is problematic for a number of reasons. For example, a Web page developer would  
18 need to create and maintain (e.g. update) several different Web pages, resulting in  
19 increased costs and the possibility of inconsistent versions. Moreover, separate  
20 indexes and links would need to be created for Web pages corresponding to various  
21 target devices, greatly increasing the sizes of current indexes and Web pages.

1 Various techniques have been developed for dynamically customizing a  
2 Web page for display by different systems. For example, style sheets allow Web  
3 page developers to define how various HTML elements appear in the context of one  
4 or more Web pages. An element is a fundamental component of the structure of a  
5 HTML document, and may include, for example, a table, a paragraph, a list, an in-  
6 line image, and the like.

7 Each element may have an associated style, including one or more  
8 formatting parameters that dictate how the element is to be displayed by a Web  
9 browser. For example, a style may include parameters directed to margins,  
10 alignment, color, size, and the like.

11 Once created, a style sheet may be applied to one or more Web pages. In the  
12 case of "cascading" style sheets (CSS), multiple style sheets may be applied to the  
13 same Web page. CSS is a well known standard developed by W3C. Currently, CSS  
14 is not supported by all Web browsers, although the standard is growing in  
15 popularity.

16 A style sheet may be linked to an HTML document by means of a LINK  
17 element:

18 <HEAD>  
19 <LINK REL=STYLESHEET HREF="style.css" TYPE="text/css">  
20 </HEAD>

21 External data files containing style information are typically identified by a ".css"  
extension, e.g., "style.css."

1 A style sheet typically includes one or more rules, which define the styles  
2 to be applied to various elements or element types before the document is  
3 displayed. A rule typically includes at least one selector and at least one style to be  
4 attached to that selector. For example, in the rule, `P {font-size: 10pt}`, the  
5 selector, `P`, is referred to as a "type" selector, and the style declaration,  
6 `{font-size: 10pt}`, represents the style to be associated with every HTML  
7 element of the type, `P` (the "paragraph" element).

8 Style sheets are normally processed on the "client side," i.e. by a Web  
9 browser, rather than on the "server side," i.e. by a Web server. The reason for this  
10 distinction lies in the fact that Web browsers include parsers, which parse the Web  
11 page into a suitable data structure, such as a parse tree. The complex manipulations  
12 required for style processing must be performed on a parse tree or the like, and  
13 parsing is a normal step in displaying a Web page by a Web browser.

14 Web servers, on the other hand, do not conventionally parse Web pages, as  
15 such is not required to deliver (serve) Web pages. Likewise, Web servers do not  
16 normally include parsers. As a result, conventional Web servers are incapable of  
17 processing style sheets.

18 Unfortunately, many Web browsers do not support style sheet processing.  
19 For example, a PDA typically has a limited memory and central processing unit  
20 (CPU). Accordingly, PDA-based Web browsers are not able to process style sheets.  
21

1 Likewise, many older Web browsers do not support style sheets, since the  
2 technology is relatively new and the standards are still in flux.

3       Accordingly, what is needed is a system and method for server-side HTML  
4 customization. What is also needed is a system and method for server-side HTML  
5 customization based on style sheets and a target device. Moreover, what is also  
6 needed is a system and method for maintaining one version of an HTML document  
7 for various types of target devices with different capabilities.





1 may flatten the DOM to generate therefrom a corresponding transformed  
2 document.

3 In one embodiment, the style sheet may be included within a separate  
4 portion of the document. In alternative embodiments, however, the style sheet and  
5 the document may comprise logically separate data files.

6 In yet another aspect of the invention, a transmission module may transmit  
7 the transformed document to a requesting client program. In various embodiments,  
8 the client program may include a Web browser.

9 These and other objects, features, and advantages of the present invention  
10 will become more fully apparent from the following description and appended  
11 claims, or may be learned by the practice of the invention as set forth hereinafter.

12

13

14

15

16

17

18

19

20

21

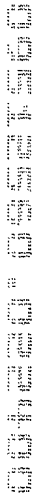
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21

Figure 1 is a schematic block diagram of a computer system suitable for hosting a plurality of software modules according to an embodiment of the invention;

Figure 3 is schematic flowchart of a method for server-side HTML customization based on style sheets and a target device according to an embodiment of the invention;

Figure 5 is an illustration of a Document Object Model (DOM) according to an embodiment of the invention;

Figure 7 is an illustration of a transformed DOM according to an embodiment of the invention;



1           Figure 8 is an illustration of a transformed HTML document according to  
2 an embodiment of the invention.

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

1                   DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

2                   Certain presently preferred embodiments of the invention are now  
3 described with reference to the Figures, where like reference numbers indicate  
4 identical or functionally similar elements. The components of the present invention,  
5 as generally described and illustrated in the Figures, may be implemented in a  
6 variety of configurations. Thus, the following more detailed description of the  
7 embodiments of the system and method of the present invention, as represented in  
8 the Figures, is not intended to limit the scope of the invention, as claimed, but is  
9 merely representative of presently preferred embodiments of the invention.

10                  Throughout the following description, various system components are  
11 referred to as "modules." In certain embodiments, the modules may be  
12 implemented as software, hardware, firmware, or any combination thereof.

13                  For example, as used herein, a module may include any type of computer  
14 instruction or computer executable code located within a memory device and/or  
15 transmitted as electronic signals over a system bus or network. An identified  
16 module may include, for instance, one or more physical or logical blocks of  
17 computer instructions, which may be embodied within one or more objects,  
18 procedures, functions, or the like.

19                  The identified modules need not be located physically together, but may  
20 include disparate instructions stored at different memory locations, which together  
21 implement the described logical functionality of the module. Indeed, a module may

1 include a single instruction, or many instructions, and may even be distributed  
2 among several discrete code segments, within different programs, and across  
3 several memory devices.

4 Figure 1 is a schematic block diagram of a computer system 10 in which a  
5 plurality of software modules may be hosted on one or more computer workstations  
6 12 connected via a network 14. The network 14 may include a wide area network  
7 (WAN) or local area network (LAN) and may also include an interconnected system  
8 of networks, one particular example of which is the Internet.

9 A typical computer workstation 12 may include a central processing unit  
10 (CPU) 16. The CPU 16 may be operably connected to one or more memory devices  
11 18. The memory devices 18 are depicted as including a non-volatile storage device  
12 20 (such as a hard disk drive or CD-ROM drive), a read-only memory (ROM) 22,  
13 and a random access memory (RAM) 24.

14 The computer workstation 12 may operate under the control of an operating  
15 system (OS) 25, such as OS/2®, WINDOWS NT®, WINDOWS®, UNIX®, and the like.  
16 In various embodiments, the OS 25 provides a graphical user interface (GUI).

17 The computer workstation 12 may also include one or more input devices  
18 26, such as a mouse and/or a keyboard, for receiving inputs from a user. Similarly,  
19 one or more output devices 28, such as a monitor and/or a printer, may be provided  
20 within, or be accessible from, the computer workstation 12.

21

1 A network interface 30, such as an Ethernet adapter, may be provided for  
2 coupling the computer workstation 12 to the network 14. Where the network 14 is  
3 remote from the computer workstation 12, the network interface 30 may include a  
4 modem, and may connect to the network 14 through a local access line, such as a  
5 telephone line.

6 Within any given computer workstation 12, a system bus 32 may operably  
7 interconnect the CPU 16, the memory devices 18, the input devices 26, the output  
8 devices 28, the network interface 30, and one or more additional ports 34, such as  
9 parallel and/or serial ports.

10 The system bus 32 and a network backbone 36 may be regarded as data  
11 carriers. Accordingly, the system bus 32 and the network backbone 36 may be  
12 embodied in numerous configurations, such as wire and/or fiber optic lines, as well  
13 as electromagnetic communication channels using visible light, infrared, and radio  
14 frequencies.

15 The computer workstations 12 may be coupled via the network 14 to one  
16 or more application servers 42, and/or other resources or peripherals 44, such as  
17 scanners, fax machines, and the like. External networks, such as the Internet 40, may  
18 be coupled to the network 14 through a router 38 or firewall.

19 In various embodiments, one or more Web servers 46 may be accessible to  
20 the workstations 12 via the Internet 40. A Web server 46 may be implemented using  
21 a workstation 12, as described above, including specialized software for delivering

(serving) Web pages to Web browsers. A variety of Web server application programs are available, including public domain software from the National Center for Supercomputing Applications (NCSA) and Apache, as well as commercial packages from Microsoft, Netscape and others.

Referring now to Figure 2, a system 48 for server-side HTML customization may include a Web server 46 and a target device 50. The target device 50 may be implemented using a workstation 12, which includes a Web browser 52, such as Netscape Navigator™ or Microsoft Internet Explorer™. The Web browser 52 may be configured to communicate with the Web server 46 via the hypertext transfer protocol ("HTTP").

In various embodiments, the target device 50 may include a standard desktop computer, such as an IBM PC™ or compatible. In alternative embodiments, however, the target device 50 may include a Web-enabled personal data assistant (PDA), such as a PalmPilot™ VII, available from 3Com Corporation, or the like.

The Web server 46 is depicted as including a request reception module 54. In one embodiment, the request reception module 54 receives (from the Web browser 52) a request for a document 56 stored within a document storage area 58 of the Web server 46. The document 56 may be encoded in the hypertext markup language ("HTML") and may include one or more HTML elements 57, as described more fully hereafter.



1 In one embodiment, the Web server 46 also includes a parsing module 60,  
2 commonly referred to as a "parser." The parsing module 60 retrieves, in various  
3 embodiments, the requested document 56 and parses the document 56 to generate  
4 therefrom a corresponding Document Object Model (DOM) 62, often referred to as  
5 a "parse tree." A DOM 62 is a tree-like, hierarchical data structure including one or  
6 more objects 64 that represent the various HTML elements 57 of the document 56.

7 In certain embodiments, the parsing module 60 is a conventional HTML  
8 parser. For example, both Netscape Navigator™ and Microsoft Internet Explorer™  
9 include HTML parsers, which may be adapted, in various embodiments, for use  
10 within the Web server 46. In an alternative embodiment, a custom HTML parser  
11 may be used. Conventionally, however, a Web server 46 does not include a parsing  
12 module 60, since a document 56 is normally parsed only by a Web browser 52 at the  
13 time the document 56 is displayed.

14 The Web server 46 may also include a style sheet access module 66. In  
15 certain embodiments, the style sheet access module 66 is configured to retrieve a  
16 style sheet 68 (from a style sheet storage area 70) including one or more rules  
17 directed to a target device 50.

18 The style sheet access module 66 may include a target device identification  
19 module 69, which may identify the type or class of the target device 50. This may  
20 be accomplished, for example, based on platform information provided as part of  
21

1 a browser request. Typically, a browser request includes a browser name and  
2 version, as well as information about the platform, such as screen resolution.

3 The style sheet access module 66 may also include a style sheet identification  
4 module 71. According to various embodiment, a single style sheet 68 may include  
5 rules directed to different target devices 50. For example, rules directed to a PDA-  
6 type device may be identified within the style sheet 68 by @media handheld  
7 indicator or the like. Consequently, the style sheet identification module 71 may  
8 identify the rules of the style sheet 68 corresponding to the identified target device  
9 50.

10 The Web server 46 may also include a style sheet application module 72,  
11 which applies the appropriate rules of the style sheet 68 to the DOM 62 of the  
12 document 56. Techniques for applying style sheets rules are well known in the art.  
13 For example, both Netscape Navigator™ and Microsoft Internet Explorer™ include  
14 style sheet application modules 72, which may be adapted, in various embodiments,  
15 for use within the Web server 46. In an alternative embodiment, however, a custom  
16 style sheet application module 72 may be used.

17 In one embodiment, the style sheet access module 66 includes an object  
18 removal module 74. Where, for instance, a rule within a style sheet 68 indicates a  
19 “NONE” display style, or similar designation, for an element 57 or element type, a  
20 corresponding object 64 within the DOM 62 is preferably removed.  
21

1 For example, the rule, `IMG { display: NONE }`, indicates a "NONE"  
2 display style for the IMG (in-line image) element type. Accordingly, the object  
3 removal module 74 preferably removes the object(s) 64 of the DOM 62  
4 corresponding to in-line image elements 57. This is advantageous, for instance,  
5 where a document 56 includes in-line images, but a target device 50, such as a PDA,  
6 cannot display such images.

7 The style sheets 68 and the Web documents 56 are depicted as logically  
8 separate data files, and may even be stored within separate storage areas 58, 70 of  
9 the Web server 46. In an alternative embodiment, a style sheet 68 may be included  
10 within a separate portion of the document 56. For example, the HTML elements 57  
11 of the document 56 and the rules of the style sheet 68 may be stored within separate  
12 portions of a single logical data file.

13 The Web server 46 may also include a flattening module 76. In various  
14 embodiments, the flattening module 76 flattens the DOM 62 to generate therefrom  
15 a corresponding transformed document 78. As used herein, the term "flattening"  
16 refers to a process of converting the DOM 62 back into an equivalent HTML  
17 document 86 including one or more corresponding HTML elements 57. Techniques  
18 for flattening a DOM 62 are well known in the art. The resulting document 86  
19 is designated as "transformed" because the style sheet application will be reflected  
20 in the HTML elements 57 of the transformed document 78.

21

1 In various embodiments, the Web server 46 may also include a transmission  
2 module 80. The transmission module 80 may send the transformed document 78  
3 (via the Internet 40) to the Workstation 12, such that the document 86 may be  
4 displayed by the Web browser 52.

5 Referring now to Figure 3, a schematic flowchart includes a method 100 for  
6 server-side HTML customization according to a presently preferred embodiment  
7 of the invention. The method 100 may begin by receiving 102, at a Web server 46,  
8 a request for a document 56.

9 Figure 4 illustrates an exemplary document 56 according to an embodiment  
10 of the invention. The document 56 may include one or more HTML elements 57,  
11 such as a paragraph element 57A and an image element 57B.

12 After the document request is received 102, the method 100 may continue  
13 by parsing 104 the document 56 to generate therefrom a corresponding Document  
14 Object Model (DOM) 62. As noted, a DOM 62 is a tree-like, hierarchical data  
15 structure including one or more objects 64 that represent the HTML elements 57 of  
16 the document 56. Figure 5 illustrates a portion of a simplified DOM 62  
17 corresponding to the document 56 of Figure 4.

18 After the document 56 is parsed 104, the method 100 may continue by  
19 identifying 106 a target device 50 for displaying the document 56. As noted, the  
20 target device 50 may be based on platform information provided by a browser  
21 request.

1 After the target device 50 is identified 106, the method 100 may continue by  
2 identifying 108 one or more rules of a style sheet 68 directed to the identified target  
3 device 50. As noted, a single style sheet 68 may include sets of rules directed to  
4 different target devices 50. For example, a rule set directed to a PDA-type device  
5 may be identified by a @media handheld indicator or the like. Consequently, the  
6 style sheet identification module 71 may identify the rules of the style sheet 68  
7 directed to the identified target device 50.

8 Figure 6 illustrates an exemplary style sheet 68 for a PDA-type target device  
9 50 according to an embodiment of the invention. The style sheet 68 may include any  
10 number of standard rules 72, such as rule-sets and at-rules (as defined in the CSS  
11 standard).

12 As previously explained, a PDA may not be capable of displaying images  
13 or other graphical objects. In addition, a PDA may be limited as to fonts, font sizes,  
14 and the like. Moreover, limited-bandwidth target devices 50, such as wireless  
15 devices, may require Web documents 56 that have reduced graphical content. The  
16 style sheet 68 may include one or more rules 72 for customizing a Web document  
17 56 for a target device.

18 For example, a first rule 73A, i.e. `P { font-size: 10pt }` may set the  
19 font size for each paragraph element 57. Specifically, the rule 73A may set the font  
20 size to 10 points.  
21

1 A second rule 73B, i.e. `IMG { display: NONE }`, may not include a  
2 typical style declaration, but may specify "NONE" display style or a similar  
3 designation. In various embodiments, a "NONE" display style causes the object  
4 removal module 74 to remove objects 64 corresponding to the element type  
5 specified in the rule 73.

6 After the style sheet 68 is identified, the method 100 may continue by  
7 applying 110 the identified style sheet rules 73 to the DOM 62. Each rule 73 of the  
8 style sheet 68 may be applied to the objects 64 of the DOM 62, which may result in  
9 the removal of certain objects 64 and the addition of others.

10 For example, as illustrated in Figure 7, the rule 73A may add a new object  
11 64E, corresponding to a `<font size=10>` element 57. By contrast, the rule 73B  
12 may cause objects 64A-C (IMG elements 57) of Figure 5 to be deleted. After  
13 application of the style sheet 68, the DOM 62 may appear as shown in Figure 7.

14 While the style sheet 68 and the document 56 are depicted herein as logically  
15 separate data files, the style sheet 68 may be included, in some instances, within a  
16 separate portion of document 56. For example, all of the rules 73 of the style sheet  
17 68 may be located, as a group, at the beginning of the document 56:

```
18 <style>  
19     P { font-size: 10pt }  
20     IMG { display: NONE }  
21 </style>  
<html>  
<head>  
<TITLE>A Simple HTML Document</TITLE>
```

1       </head>  
2       <body>

3       . . .

4       In alternative embodiments, a single style sheet 68 may include portions  
5       corresponding to two or more target devices 50. For example, a style sheet 68 may  
6       include the following:

7       @media handheld  
8       {  
9       P { font-size: 10pt }  
10      IMG { display: NONE }  
11      }  
12      @media tinyscreen  
13      {  
14      P { font-size: 12pt }  
15      IMG { display: NONE }  
16      }

17       In such an embodiment, the style sheet access module 66 may parse the style sheet  
18       68 and extract the rules 73 corresponding to the identified target device 50.

19       After the rules 73 have been applied 110, the method 100 may continue by  
20       flattening 112 the DOM 62 to create a transformed document 78, which may then  
21       be sent 114 to the requesting Web browser 52 for display. As previously noted, the  
flattening process involves converting the DOM 62 back into an HTML document  
86. Consequently, any transformations to the DOM objects 64 will be preferably  
reflected in the corresponding HTML elements 57 of the document 86.

For example, Figure 8 illustrates an exemplary transformed document 78  
after flattening 116 the DOM 62 of Figure 7. Comparing the transformed document

1 78 of Figure 8 to the requested document 56 of Figure 4 reveals that a new HTML  
2 element 57C is added, and the image elements 57 of Figure 4, including element 57B,  
3 are deleted.

4 Based on the foregoing, the present invention offers a number of advantages  
5 not found in conventional approaches. Style sheets 68 are processed on the server  
6 side, which is advantageous for target device 50 that are not capable of style sheet  
7 processing, such as PDAs.

8 Moreover, the system and method of the present invention make it possible  
9 to maintain one version of a Web document 56 for a variety of target devices 50,  
10 each of which may have different capabilities. Thus, different target devices 50 may  
11 access a Web document 56 using the same URL, which minimizes development and  
12 maintenance costs and the need for multiple links for different target devices 50.

13 Even target devices 50 that are capable of processing style sheets 68 may  
14 benefit from the present invention, such as those with a limited bandwidth (e.g.  
15 wireless devices). Because style sheets 68 are conventionally applied by a Web  
16 browser 52, a wireless target device 50 must first retrieve a document 56 and a  
17 corresponding style sheet 68 before the style sheet 68 may be applied.  
18 Unfortunately, if the document 56 is large, the bandwidth has already been wasted.

19 By contrast, the system and method of the present invention apply style  
20 sheets 68 on the Web server 46. Server-side HTML customization results in a more  
21 compact document 56 that may be sent to a target device 50 over a limited-



1 bandwidth network. Moreover, the need for bandwidth is further reduced because  
2 the style sheets 68 are never sent to the target device 50.

3 The present invention may be embodied in other specific forms without  
4 departing from its scope or essential characteristics. The described embodiments  
5 are to be considered in all respects only as illustrative and not restrictive. The scope  
6 of the invention is, therefore, indicated by the appended claims rather than by the  
7 foregoing description. All changes which come within the meaning and range of  
8 equivalency of the claims are to be embraced within their scope.

9 What is claimed is:  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21

Claims

1. Within a document server, a computer-implemented method for customizing a requested document comprising at least one hypertext markup language (HTML) element, the method comprising:

parsing the document to generate therefrom a corresponding document object model (DOM) including at least one object;  
obtaining a style sheet including at least one rule directed to a target device;  
applying the at least one rule of the style sheet to the DOM; and  
flattening the DOM to generate therefrom a corresponding transformed document suitable for display by the target device.

2. The method of claim 1, wherein the style sheet comprises a cascading style sheet (CSS).

3. The method of claim 1, wherein the obtaining step comprises:  
identifying a target device for displaying the document; and  
identifying at least one rule of a style sheet directed to the identified target device.

4. The method of claim 3, further comprising:

1 receiving a request for a document from a client program.

2  
3 5. The method of claim 4, wherein the client program comprises a Web  
4 browser.

5  
6 6. The method of claim 1, wherein the style sheet includes rules directed  
7 to at least two different target devices.

8  
9 7. The method of claim 1, wherein the style sheet is stored within a  
10 separate portion of the document.

11  
12 8. The method of claim 1, wherein the style sheet and the document are  
13 stored as logically separate data files.

14  
15 9. The method of claim 1, further comprising:  
16 transmitting the transformed document to a client program.

17  
18 10. The method of claim 1, the transforming step comprising:  
19 removing at least one object of the DOM in response to an indication within  
20 the style sheet to remove a corresponding HTML element from the  
21 document.

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

11. A system for customizing a requested document comprising at least one hypertext markup language (HTML) element, the system comprising:

a parsing module configured to parse the document to generate therefrom

a corresponding document object model (DOM) including at least one

object;

a style sheet access module configured to obtain a style sheet including at

least one rule directed to a target device;

a style sheet application module configured to apply the at least one rule of

the style sheet to the DOM; and

a flattening module configured to flatten the DOM to generate therefrom a

corresponding transformed document suitable for display by the

target device.

12. The system of claim 11, wherein the style sheet comprises a cascading style sheet (CSS).

13. The system of claim 11, wherein the style sheet access module comprises:

a target device identification module configured to identify a target device

for displaying the document; and

1 a style sheet identification module configured to identify at least one rule  
2 within a style sheet directed to the identified target device.  
3

4 14. The system of claim 13, further comprising:  
5 a request reception module configured to receive a request for a document  
6 from a client program.  
7

8 15. The system of claim 14, wherein the client program comprises a Web  
9 browser.  
10

11 16. The system of claim 11, wherein the style sheet includes rules directed  
12 to at least two different target devices.  
13

14 17. The system of claim 11, wherein the style sheet is stored within a  
15 separate portion of the document.  
16

17 18. The system of claim 11, wherein the style sheet and the document are  
18 stored as logically separate data files.  
19

20 19. The system of claim 11, further comprising:  
21

1 a transmission module configured to transmit the transformed document to  
2 a client program.

3  
4 20. The system of claim 11, wherein the style sheet application module  
5 comprises:

6 an object removal module configured to remove at least one object of the  
7 DOM in response to an indication within the style sheet to remove a  
8 corresponding HTML element from the document.

9  
10 21. An article of manufacture comprising a program storage medium  
11 readable by a processor and embodying one or more instructions executable by the  
12 processor to perform a computer-implemented method for customizing a requested  
13 document comprising at least one hypertext markup language (HTML) element, the  
14 method comprising:

15 parsing the document to generate therefrom a corresponding document  
16 object model (DOM) including at least one object;  
17 obtaining a style sheet including at least one rule directed to a target device;  
18 applying the at least one rule of the style sheet to the DOM; and  
19 flattening the DOM to generate therefrom a corresponding transformed  
20 document suitable for display by the target device.

21

22. The article of manufacture of claim 21, wherein the style sheet comprises a cascading style sheet (CSS).

23. The article of manufacture of claim 21, wherein the obtaining step comprises:

identifying a target device for displaying the document; and

identifying at least one rule of a style sheet directed to the identified target device.

24. The article of manufacture of claim 23, the method further comprising: receiving a request for a document from a client program.

25. The article of manufacture of claim 24, wherein the client program comprises a Web browser.

26. The article of manufacture of claim 21, wherein the style sheet includes rules directed to at least two different target devices.

27. The article of manufacture of claim 21, wherein the style sheet is stored within a separate portion of the document.

1           28.    The article of manufacture of claim 21, wherein the style sheet and the  
2 document are stored as logically separate data files.

3  
4           29.    The article of manufacture of claim 21, the method further comprising:  
5 transmitting the transformed document to a client program.

6  
7           30.    The article of manufacture of claim 21, the transforming step  
8 comprising:  
9 removing at least one object of the DOM in response to an indication within  
10 the style sheet to remove a corresponding HTML element from the  
11 document.



# SERVER-SIDE HTML CUSTOMIZATION BASED ON STYLE SHEETS AND A TARGET DEVICE

## ABSTRACT OF THE DISCLOSURE

A request reception module receives a request for an document stored within document server. A parsing module parses the document to generate therefrom a corresponding document object model (DOM) including at least one object. A style sheet access module obtains a style sheet including at least one rule directed to a target device for displaying the document. An style sheet application module applies the at least one rule of the style sheet to the DOM. A flattening module flattens the DOM to generate therefrom a corresponding transformed document. A transmission module transmits the transformed document to a requesting client program.

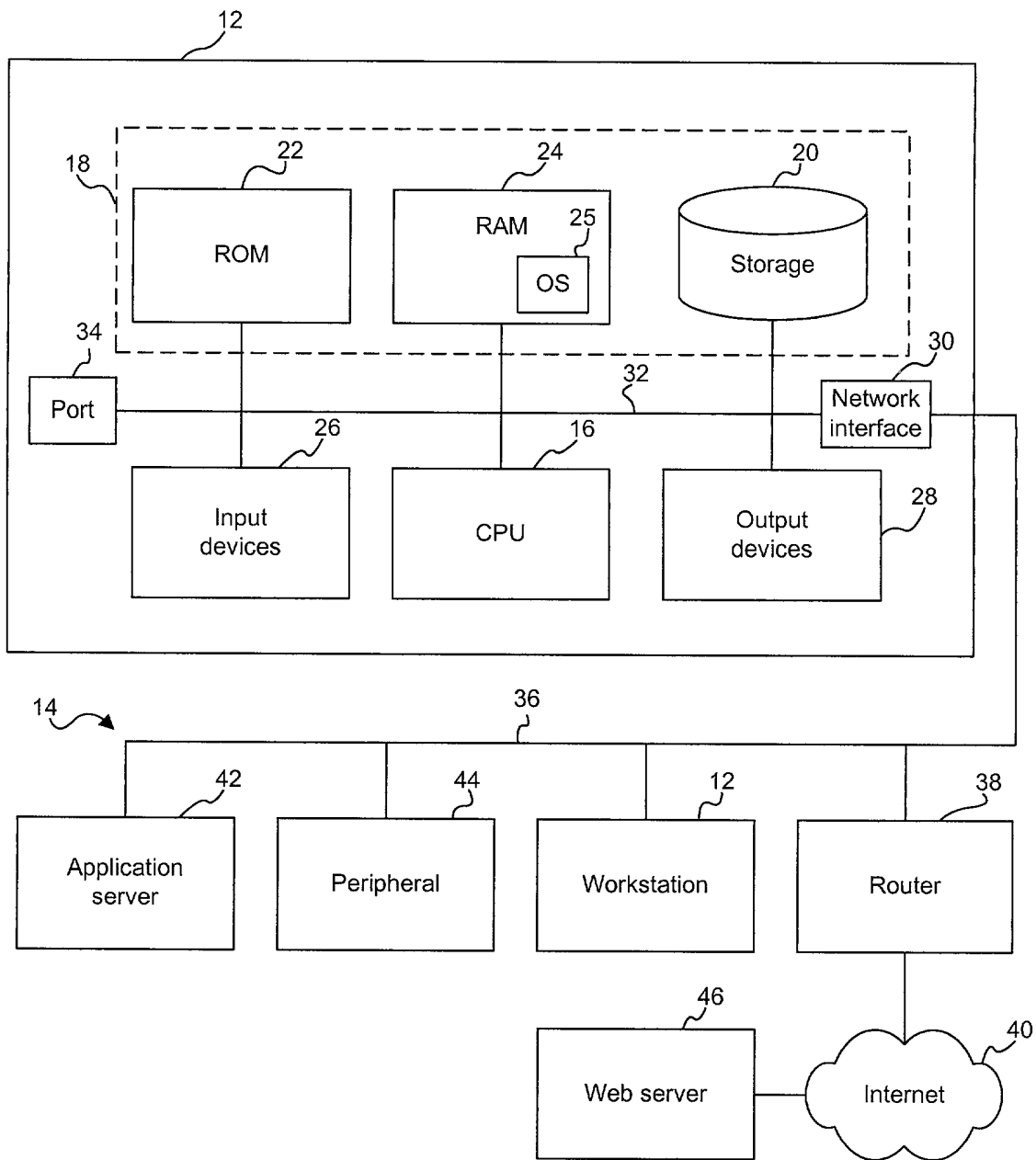


Fig. 1

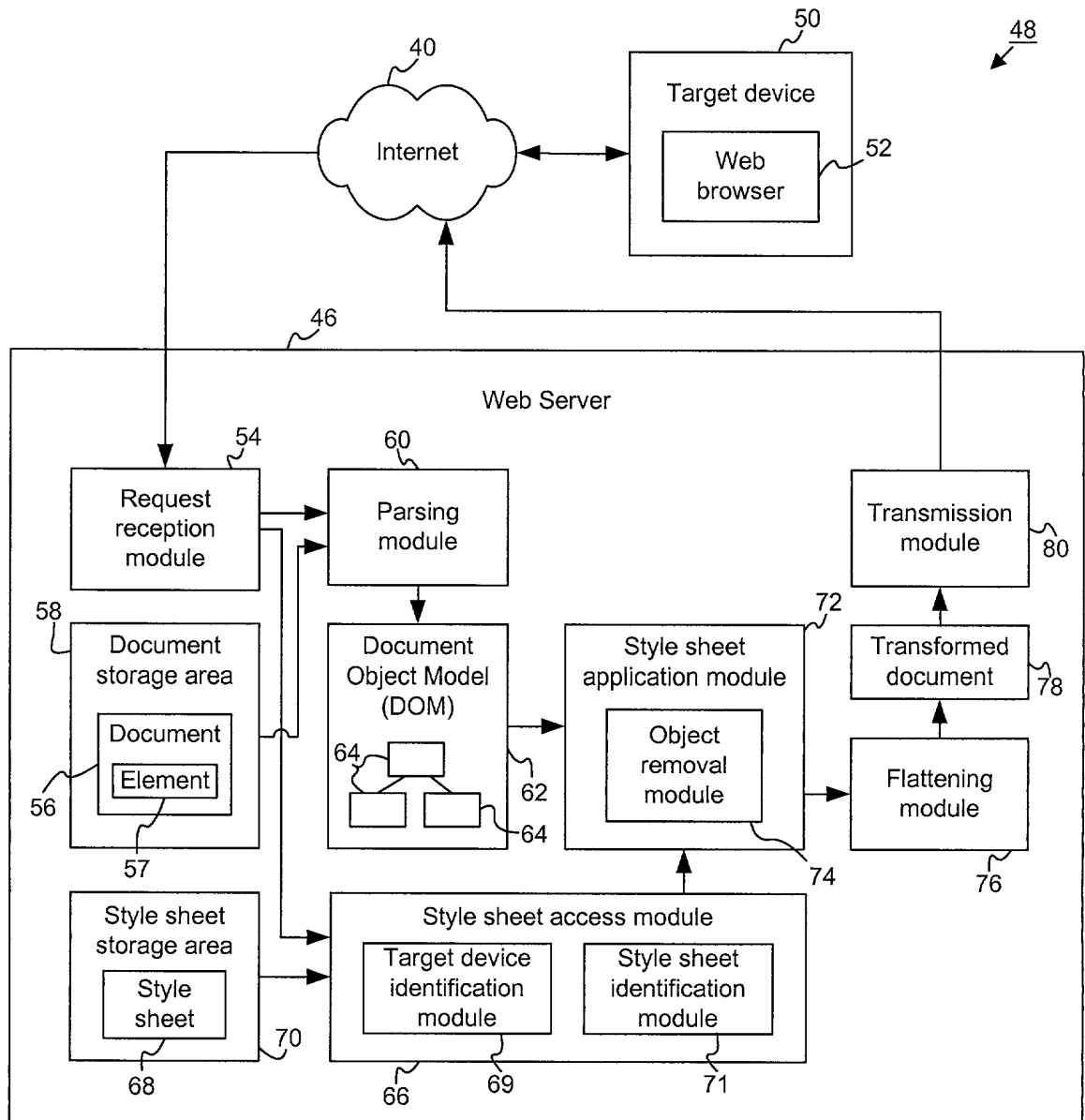


Fig. 2

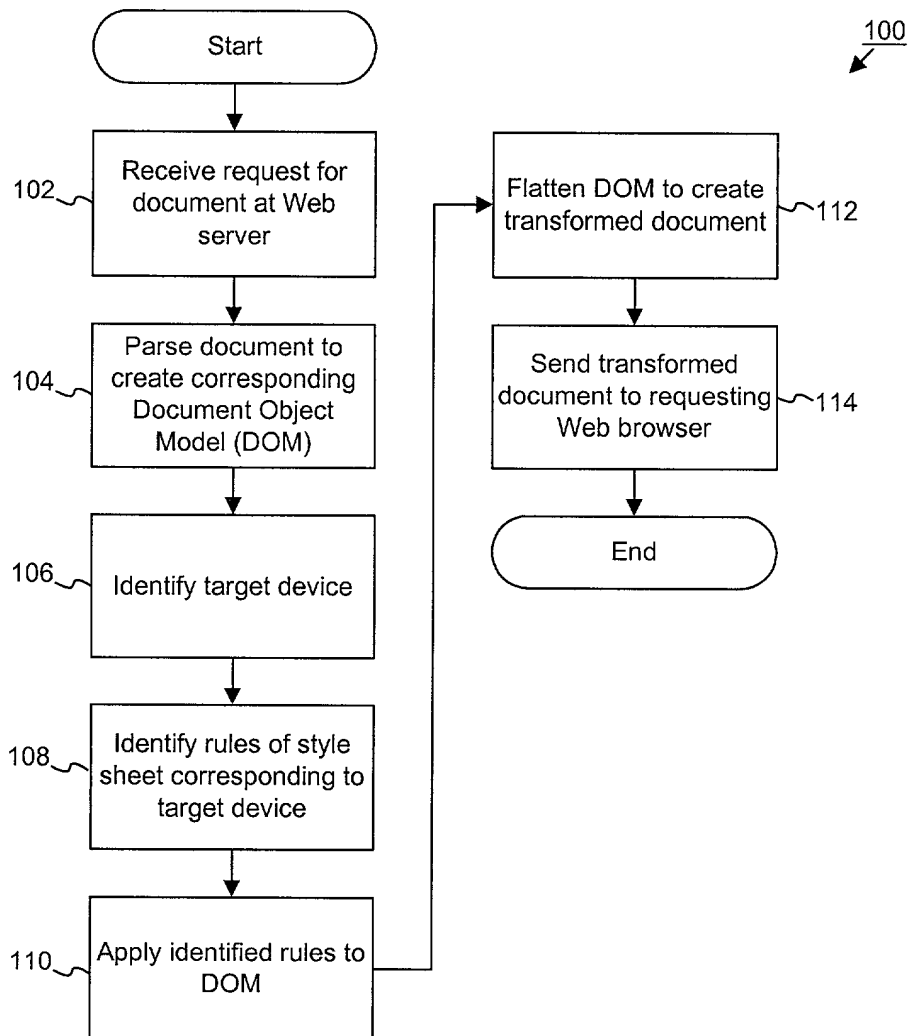


Fig. 3

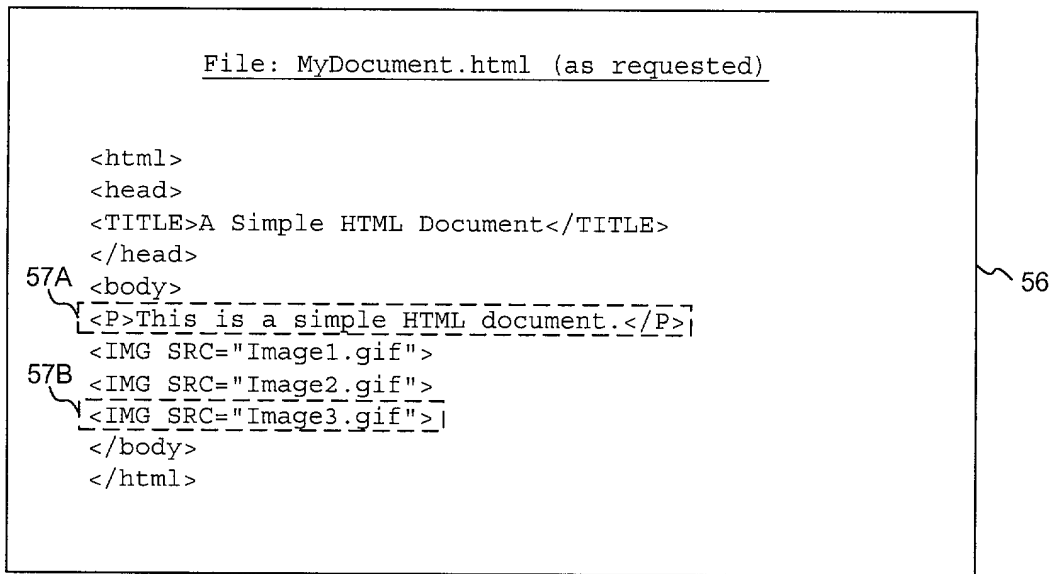


Fig. 4

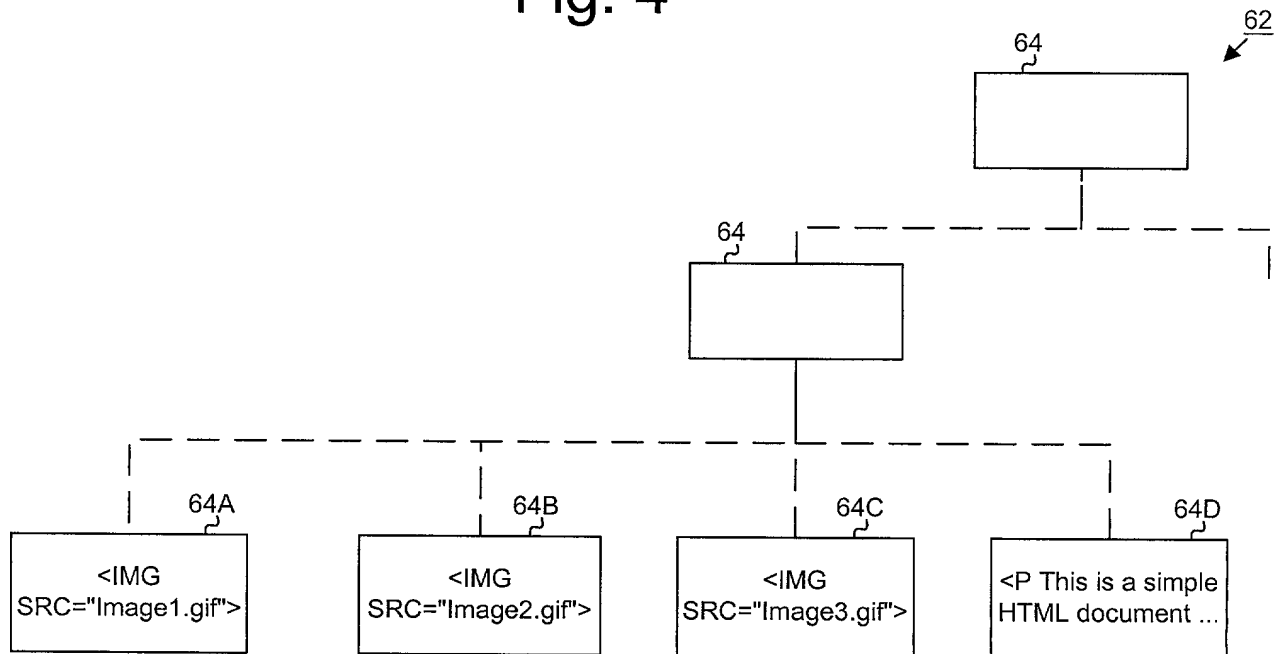


Fig. 5

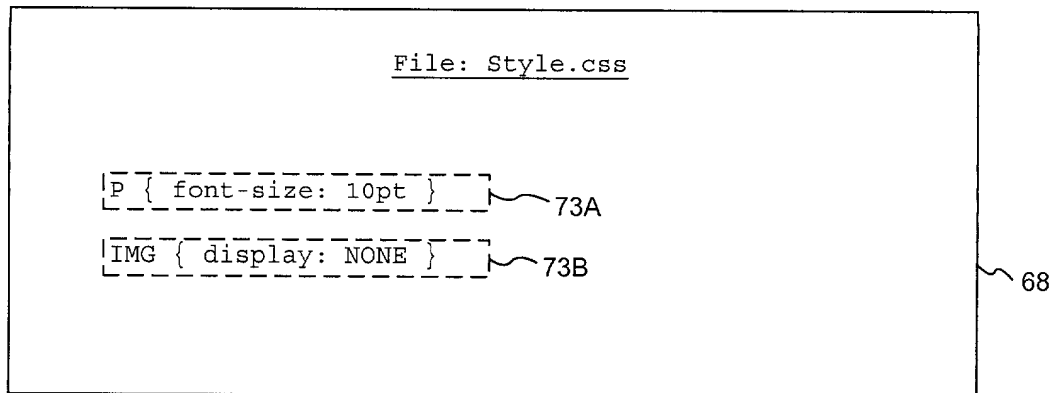


Fig. 6

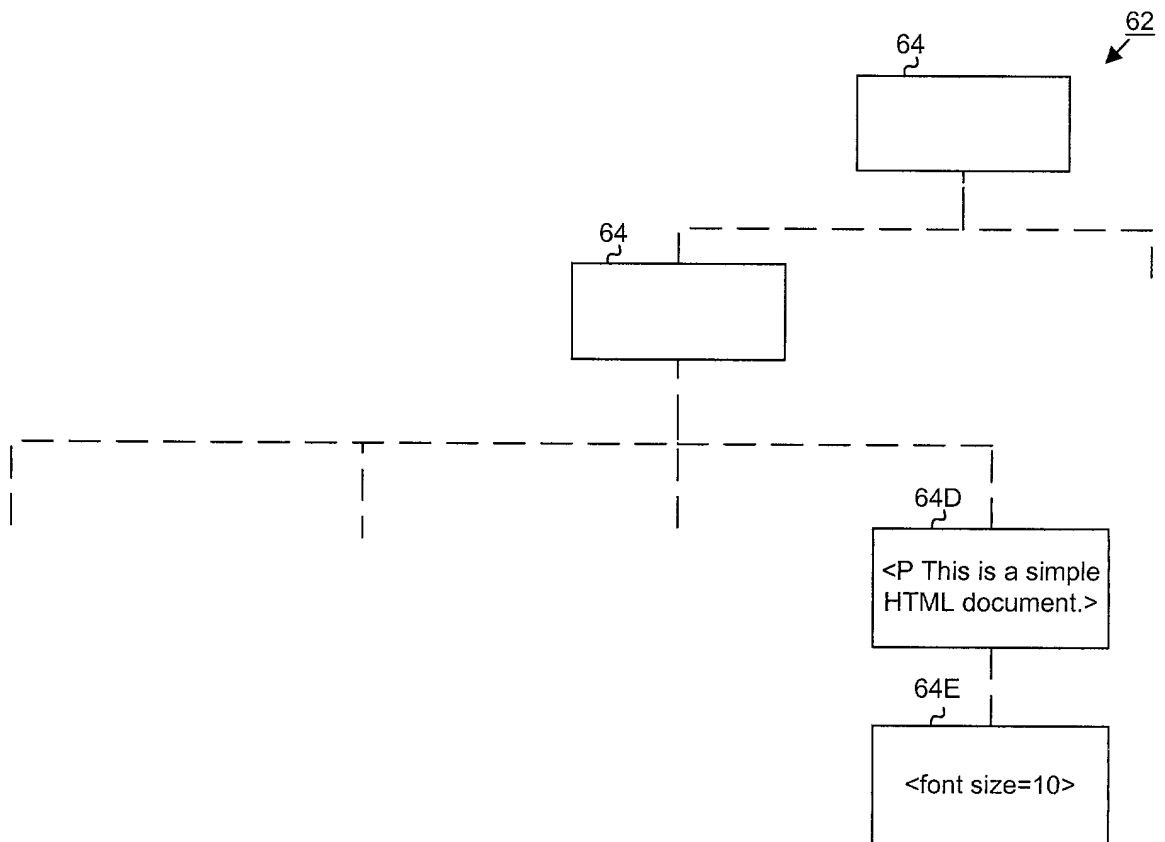


Fig. 7

File: MyDocument.html (as transmitted)

```
<html>
<head>
<TITLE>A Simple HTML Document</TITLE>
</head>
<body>
<P><font size=10>This is a simple HTML document.</P>
</font>
</body>
</html>
```

78

57C

Fig. 8

**DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION**

As a below named inventor, I hereby declare that:

My residence and citizenship are as stated below next to my name;

I believe I am the original, first and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled

SERVER-SIDE HTML CUSTOMIZATION BASED ON STYLE SHEETS AND A TARGET DEVICE

the specification of which (check one)

  X   is attached hereto.  
       was filed on \_\_\_\_\_  
as Application Serial No. \_\_\_\_\_  
and was amended on \_\_\_\_\_ (if applicable).

I hereby stat that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, united States Code, Section 119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)			Priority Claimed
<u>  none  </u>	<u>                    </u>	<u>                    </u>	<u>      </u> Yes <u>      </u> No
(Number)	(Country)	(Day/Month/Year filed)	

I hereby claim the benefit under Title 35, Untied States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56, which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

<u>  none  </u>	<u>                    </u>	<u>                    </u>
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

OFFICE OF THE SECRETARY OF COMMERCE



**POWER OF ATTORNEY:** As named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith. (list name and registration number)

Romualda Strimaitis: 35,697  
Timothy M. Farrell: 37,321  
Ingrid M. Foerster: 36,511  
Prentiss W. Johnson: 33,123  
Christopher A. Hughes: 26,914  
John E. Hoel: 26,279  
Edward A. Pennington: 32,588  
Joseph C. Redmond, Jr.: 18,753  
Craig J. Madson: 29,407  
L. Craig Metcalf: 31,398

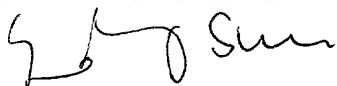
Evan R. Witt: 32,512  
A. John Pate: 36,234  
Gary D.E. Pierce: 38,019  
David B. Fonda: 39,672  
John R. Thompson: 40,842  
Brian C. Kunzler: 38,527  
Barton W. Giddings: 41,036  
Hal D. Baird: 42,284  
Kory D. Christensen: 43,548

Send correspondence to:

Kory D. Christensen  
MADSON & METCALF  
15 West South Temple, Suite 900  
Salt Lake City, Utah 84101  
Telephone: (801) 537-1700

Full name of sole or first joint-inventor: Yudong Sun

Inventor's signature:



Date:

2/16/2000

Residence: 6340 Snowberry Court, Gilroy, California 95020

Citizenship: China

Post Office Address: Same